symmetry

Article

TCAN-IDS: Intrusion Detection System for Internet of Vehicle Using Temporal Convolutional Attention Network

Pengzhou Cheng, Kai Xu, Simin Li and Mu Han *

School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China; 2221908010@stmail.ujs.edu.cn (P.C.); 2222008083@stmail.ujs.edu.cn (K.X.); 2222008032@stmail.ujs.edu.cn (S.L.)
* Correspondence: hanmu@ujs.edu.cn

Abstract: Intrusion detection systems based on recurrent neural network (RNN) have been considered as one of the effective methods to detect time-series data of in-vehicle networks. However, building a model for each arbitration bit is not only complex in structure but also has high computational overhead. Convolutional neural network (CNN) has always performed excellently in processing images, but they have recently shown great performance in learning features of normal and attack traffic by constructing message matrices in such a manner as to achieve real-time monitoring but suffer from the problem of temporal relationships in context and inadequate feature representation in key regions. Therefore, this paper proposes a temporal convolutional network with global attention to construct an in-vehicle network intrusion detection model, called TCAN-IDS. Specifically, the TCAN-IDS model continuously encodes 19-bit features consisting of an arbitration bit and data field of the original message into a message matrix, which is symmetric to messages recalling a historical moment. Thereafter, the feature extraction model extracts its spatial-temporal detail features. Notably, global attention enables global critical region attention based on channel and spatial feature coefficients, thus ignoring unimportant byte changes. Finally, anomalous traffic is monitored by a two-class classification component. Experiments show that TCAN-IDS demonstrates high detection performance on publicly known attack datasets and is able to accomplish real-time monitoring. In particular, it is anticipated to provide a high level of symmetry between information security and illegal intrusion.

Keywords: control area network; intrusion detection system; temporal convolution network; attention mechanism

check for updates

Citation: Cheng, P.; Xu, K.; Li, S.; Han, M. TCAN-IDS: Intrusion Detection System for Internet of Vehicle Using Temporal Convolutional Attention Network. *Symmetry* **2022**, *14*, 310. https://doi.org/10.3390/sym14020310

Academic Editor: Basil Papadopoulos

Received: 13 January 2022
Accepted: 27 January 2022
Published: 3 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

In recent years, the traditional mechanical interface to the vehicle system has been replaced by communication between electronic control units (ECUs) or minicomputers that coordinate subsystems [1]. Each ECU is the implementer of a specific function in the control system, such as throttle, brake, and steering. The ECUs are segmented into different subnets according to their function or communication rate. The control area network (CAN) of in-vehicle interconnects the subnets via multiple gateways, providing an efficient, reliable, and economical communication channel. The vehicle ecosystem is moving towards being internet-connected, intelligent, and modernized [2]. At the same time, the Internet of Vehicles (IoV) has enabled the world to witness a deluge of data that is building a new product of the Internet of Everything.

The CAN has been a de facto standard for serial communication in vehicle netwrok. Unfortunately, as the in-vehicle network did not communicate with external networks at the time, the CAN bus neglected security considerations in its design [3]. The broadcast mechanism of the CAN protocol and the unauthenticated and unencrypted communication of messages all evolved into potential security problems [4]. With the rapid development of intelligent networked vehicles (ICV), there are various ports in vehicles that can be connected to internal and external devices, particularly the intelligent sensing systems deployed on the cloud and edge devices [5]. Meanwhile, it also opens the door for attackers to launch attacks to make their desired decisions. The aforementioned security vulnerabilities motivate adversary to compromise a vehicle effortlessly through various method, such as remote attacks, infotainment system attacks, sensor attacks, and direct interface attacks [6]. Moreover, the channels of attack include Bluetooth, USB flash drives, temperature sensors, and onboard diagnostics (OBD-II) ports [3,7–9]. The frightening part is that, in addition to disrupting normal communications in vehicles, they also enable malicious listening in order to obtain sensitive and valuable data. However, these valuable data are generally localized in practical applications and divided into asymmetric and symmetric data distributions. Examples include symmetrical relationships between data from extra-vehicular networks and asymmetrical probability distributions of malicious and normal network traffic [10]. Hence, the information security problem has always been one of the important factors of ICV that should be considered [2,11].

At present, intrusion detection systems (IDSs) are defensive systems that are being tasked with the critical task of detecting malicious cyber attacks [12]. In particular, artificial intelligence is increasingly applicable in various of engineering fields [13–15]. Hence, the huge amount of network data makes the intrusion detection problem suitable for deep learning (DL) methods [16]. These methods have been successfully applied to database protection [17], Unmanned Aerial Vehicle (UAV) networks [18], and some famous sensor networks [19]. Hence, many researchers have worked on building IDS to identify malicious network attacks in order to secure Internet of Vehicle in the last decade [20,21]. In general, DL-based in-vehicle intrusion detection is essentially a feature extraction and classification problem. Features are extracted from the packets of the in-vehicle network with the aim of reducing data complexity and constructing valuable information, and then feature information is utilized to classify normal and attack classes. Feature extraction and classification methods can currently be divided into three main categories: convolutional neural networks (CNN), recurrent neural networks (RNN), and temporal convolutional networks (TCN).

Initially , the researchers endeavoured to effectively model an IDS of internet of vehicle (IoV) based on time series [22]. This was because denial of service (DoS) attacks, man-in-the-middle (MIMT) attacks, etc., all corrupted the original chronological sequence of normal CAN communication, i.e., the arbitration bit of CAN. For example, the IDS built by Song et al. [11] based on the arbitration bits of CAN is apparently very effective against DoS attacks with anomalous frequencies. Thereafter, fuzzing and spoofing attacks emerged, causing researchers to shift their attention to monitoring the data domain. The IDS proposed by Qin et al. [2] based on CAN data bits is apparently very effective against above attacks. More recently, IDSs have been designed to resist more attacks rather than only one of them. As a result, the idea of spatial-temporal features has shown greater advantages in the field of anomaly detection [23–25]. Accurate mapping of spatial-temporal details is one of the effective methods to identify frequency anomalies and data command anomalies. However, the IDS design inevitably requires the use of both time-series (e.g., RNN) and convolutional networks, which results in a complex network structure and sluggish inference speed [26]. As a supplement, spatial-temporal characteristics capturing important information over the entire communication sequence is one of the influences to be considered by IDS.

In order to overcome the above problems, we propose TCAN-IDS, an intrusion detection method on controller area network based on the temporal convolutional network (TCN) [27]. The TCN model has been shown to outperform time-series neural networks on multi-featured temporal data. It is able to extract better temporal and spatially local features based on CAN messages. On top of this, we believe it is important to focus on more valuable information through global attention [28]. As a result, our model will be able to extract effective spatial-temporal detail features. Afterwards, we utilize a binary classification model to discriminate anomalous traffic, which will help improve model detection performance and obtain a lower false-positive rate.

This paper presents the following contributions:

1. We developed TCAN-IDS; a novel intrusion detection model based on the TCN network. It can detect various known attacks using the temporal and spatial features of CAN images.
2. The introduction of a global attention mechanism allows the residual convolution component to pay more attention to channel and spatial information, enabling accurate mapping of features and more effective attention to important regions in the CAN image;
3. Testing the TCAN-IDS model on a real dataset, the evaluation metrics were compared to the other three baseline models and the results showed that we obtained high or the same detection performance from the best baseline approach for all.

The remaining parts of this paper are organized as follows. Section 2 reviews the work related to anomaly detection on in-vehicle networks. Section 3 presents knowledge about the CAN bus and common types of attacks. Section 4 presents model specific design details, and Section 5 provides performance evaluation. Finally, we conclude this study.

2. Related Work

The section reviews several recently proposed algorithms for CAN bus intrusion based on deep learning (DL). On the basis of the detection range, the intrusion detection algorithms are divided into those based on periodic CAN arbitration bits, data domain, and considering both arbitration bits and data domain. Moreover, we discussed the difference of existing methods to present our advantages.

2.1. In-Vehicle Intrusion Detection

Periodicity-based anomaly detection in the arbitration domain is an analysis into the frequency characteristics of message transmissions. In other words, these frequency anomalies are network packets that appear at inappropriate time locations and disrupt the availability and integrity of CAN communications. Generally, these are referred to as temporal attacks because they are highly time dependent [29]. Song et al. [30] proposed generative adversarial network-based intrusion detection system (GAN). Their method using two discriminators can learn frequency features of CAN arbitration bit and then detect unknown attack. Similarly, Song et al. [11] improved the Inception-ResNet model to build 29 × 29 2-D inputs in the deep convolution neural network. Compared with traditional machine-learning algorithms, their method has low false-negative rates and error rates. Recently, Han et al. [31] designed an IDS based on three components, including encoder, processor, and decoder, to detect both known and unknown attacks. They achieved better detection results on DoS attacks.

In contrast, data domain-based anomaly detection mines the 8 bytes or 64 bits data features of CAN message transmission. Taylor et al. [32] introduced an IDS that feed al the 64-bit of the CAN data field into the long-short term network (LSTM), and then the optimized model reduces error in predicting CAN for n + 1 messages. On this basis, Qin et al. [2] proposed two detection models, i.e., inputting the content of the data domain in hexadecimal and binary to the LSTM model, respectively. The predictions were performed separately on different CAN IDs by means of improved loss functions, and high accuracy was obtained. A further improvement is the deep LSTM network proposed by Pawelec et al. [1]. They observed the content of the first 10 data fields in order to predict the class of the 11th message. Their method verifies that data with dependence on prior or concurrent inputs are very promising.

In order to improve the generalization of the IDS, the extraction of spatial-temporal detail features is considered a necessary measure, as this accommodates both temporal and content attacks. Tariq et al. [24] presented CANTransfer, a transfer learning-based intrusion detection system that builds a convolutional LSTM-based model on known CAN message to detect unknown attacks. However, the selection of more features makes the model's detection performance on known attacks relatively low. An IDS integrating CNN and RNN was proven to be complementary to identify anomalies by Yue et al. [26]. They achieved a high accuracy rate based on building an integrated classifier with dynamic weights on several typical CNN and RNN models, but the complexity of the model was relatively high. Sun et al. [23] introduced another model of IDS based on convolutional LSTM networks. Their model extracts each time-step feature sequentially and then focuses on the important time-steps based on attention, thus improving the convergence speed and prediction accuracy of the model. In addition, the temporal convolutional networks (TCNs) have been shown to be significantly superior in spatial-temporal detail. Yue et al. [29]. established a basic detection model based on TCN that is adept at handling spatial-temporal features. The dynamic neural network technique they used optimizes the basic detection model, reducing computational consumption and improving detection capabilities.

2.2. Methods Comparison

Although various research has been published on the development of intrusion detection for in-vehicle networks, most have been designed for specific domains of frames and are, therefore, only effective for specific attacks. The differences in the latest research are shown in Table 1. Efforts to model CAN ID all capture temporal features to some extent [11,30,31], and some IDS algorithms built on data domains all exhibit different detection capabilities for data content anomalies [1,2,32]. Moreover, a plurality of papers [24,26,29] are dedicated to mining spatio-temporal detail features to resist more attacks. The intrusion detection system proposed in [23] is the only approach that considers both spatial-temporal detail features and captures important information through attention.

Table 1. A comparison of current intrusion detection techniques in-vehicle network.

Paper	Temporal Feature	Content Feature	Spatial-Temporal Feature	Attention Mechanism
Song et al. [30]	√			
Song et al. [11]	√			
Han et al. [31]	√			
Taylor et al. [32]		√		
Qin et al. [2]		√		
Pawelec et al. [1]		√		
Tariq et al. [24]			√	
Yue et al. [26]			√	
Sun et al. [23]			√	√
Yue et al. [29]			√	
TCAN-IDS			√	√

The spatial-temporal detail feature is an accompanying convolutional structure during state-to-state transitions to obtain spatial information, outperforming fully connected LSTM, convolutional networks, or integrated models. The application of attention allows for accurate feature selection during the modeling, thereby discarding manually defined features prior to training. Thus, there should still be a requirement to design an IDS for extracting valuable spatial-temporal detail features and then searching for significant changes in the feature space through an attention mechanism. Our proposed method aims to achieve this goal.

Compared with previous works, the TCAN-IDS model designed in this paper is used temporal convolution network (TCN) to mine spatial-temporal detail features. In addition, the residual component in the TCN-IDS model uses a convolutional component with global attention instead of a normal convolutional layer. This will improve the model's ability to ignore invariant features with multiple features, thus paying more attention to features that change frequently and increasing the speed of model inference.

3. Preliminaries

3.1. Control Area Network

The CAN bus network, which is a message-based broadcast system intended to allow ECUs to interact with one another, is used in modern vehicles. Depending on the length of the arbitration field, the frame standard for CAN bus transmissions is separated into two versions: the 11-bit 2.0A standard frame format and the 29-bit 2.0B extended frame format. Figure 1 illustrates the structure of a CAN data frame.

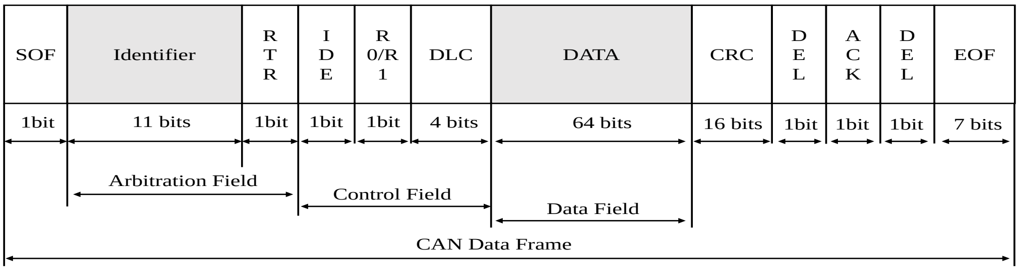

Figure 1. Illustrates the structure of the CAN standard frame. Despite the fact that extended frames have differences in arbitration bits, it is compatible with standard frame data.

The intrusion detection model proposed in this paper takes into account both the arbitration field and the message load field of the CAN message in order to resist injection attacks owing to frequency or data anomalies. the arbitration field consists of 11 bits (29 bits in extended mode) and is used to identify the message. In addition, the arbitration field indicates the frame priority of the arbitration, with the lowest value indicating the highest priority. The data load field transmits the value of the in-vehicle signal which is no larger than 8 bytes and the length is labeled by data length code (DLC).

3.2. Attack Model

In this paper, we use the "CAN Signal Extraction and Translation Dataset" provided by the Hacking and Countermeasures Research Laboratory (HCRL) [30]. The CAN protocol uses a contested bus approach to message transmission, in which each ECU node can randomly request the use of the bus. In addition, the ECU broadcasts messages to the CAN bus, which are all avenues for attackers to launch their attacks.

In Figure 2, we present the consequences of an attack on the CAN bus. It can be observed that attackers can achieve injection attacks through both remote communication devices and direct interfaces such as the OBD-II port. Once there is an ECU compromised, the attacker will maliciously sniff the data or even implement a reverse to input more precise malicious commands to control the vehicle. The details of these attack models are illustrated as follows:

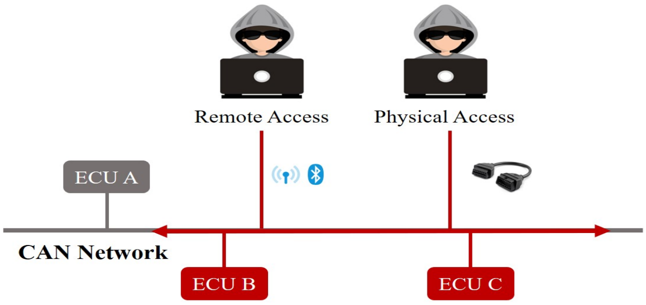

Figure 2. The process by which an attacker could compromise an in-vehicle network.

DoS attack: DoS attack aim is to send a large number of legitimate requests that exceed the system's processing capacity in order to cripple or interfere with the intended functionality of the system, which will eventually exhaust the system's resources and bring functionality to a halt. Attackers typically inject high priority messages or execute attacks using the same CAN message arbitration and known ECU transmission rates.

Fuzzy attack: This attack is stealthier and, hence, detecting anomalies is more difficult. The CAN IDs in the injected CAN frames are generated randomly from 0x000 to 0x7ff, and the data fields are usually forged as well.

Spoofing attack: The attacker simulates the frequency of messages sent by the victim node and then forges data to send anomalous messages to the CAN bus. This injection pattern will dangerously alter the state of the vehicle. Gear and RPM type spoofing attacks are shown in the dataset.

These attack models are the most common and are widely considered in the literature evaluating IDS models. In this study, our goal is to extract local and temporal features of multidimensional CAN frames to implement a lightweight, real-time intrusion detection system.

3.3. CAN Traffic Analysis

In order to help the TCAN model to better exploit the characteristics of CAN messages, we carefully analyzed the relationship between changes in the time and space domains of CAN messages. Table 2 illustrates a selection of CAN packet samples from the public dataset, where the first column represents the timeline to the end of the CAN message; the second column is the sender arbitration field of the message; the number of occupied data field bytes is shown in the third column, and the data field is shown in the fourth column.

Table 2. Sample CAN packets with various ID and DLC fields.

Timestamp	ID	DLC	Data
1478198376.389427	0316	8	05, 21, 68, 09, 21, 21, 00, 6f
1478198376.389636	018f	8	fe, 5b, 00, 00, 00, 3c, 00, 00
1478198376.389864	0260	8	19, 21, 22, 30, 08, 8e, 6d, 3a
1478198376.390096	02a0	8	64, 00, 9a, 1d, 97, 02, bd, 00
1478198376.409484	05f0	2	01, 00

We note that the data fields of the CAN messages are not uniform, which will be described in detail in Section 4.1. In addition, we observed that the semantics of bus-transmitted messages are difficult to understand, but the deep learning-based approach does not require understanding, rather than looking directly for specific patterns in the communication traffic. We statistically find that there is a finite number of sensor IDs on the vehicle, and they all broadcast data for a certain period of time to maintain the steady state of the vehicle, which reflects a clear time-series characteristics. Moreover, we also discover that there are bytes on the data field that change frequently and those that remain constant; for example, in the gear sensor-0x43F (10 50 60 FF 46 28 0A 00) only the sixth byte changes frequently, while the rest of the bytes remain unchanged. As a result, these patterns are extremely effective for simulating the usual behavior of CAN packets. At the same time, this motivates us to direct our attention to more valuable information. This permits us to detect infractions and identify the intended traffic.

4. Implementation

In this section, we explain how the TCAN-IDS model works. TCAN-IDS, similarly to other machine learning-based intrusion detection system, consists of two steps: training and detection. While a dump of CAN traffic with labels is utilized in the training stage, genuine CAN traffic without labels is used directly in the detection step in the proposed IDS technique. The next step-by-step process of building the model is described as data pre-processing, spatial-temporal feature extraction, and global attention building.

4.1. Pre-Processing Data

Qin et al. [2] proposed to utilize all data fields where eight bytes are divided into 16 numbers as 16 hexadecimal feed into LSTM network, but they did not consider the feature of CAN ID. In other words, many intruders launch types of attacks such as DoS and Replay on ECU frequencies; thus, if the model can focus on CAN ID characteristics, it will quickly identify anomalous target traffic. In this paper, we consider the CAN ID as a 3-bit feature that, together with the 16-bit data feature, forms the input to the model.

Due to the fact that the attack dataset was obtained in its rawest state, we need to first perform data pre-processing, which is divided into missing value filling, feature selection and splitting, and transformation into a two-dimensional (2-D) CAN matrix, as described in Algorithm 1.

Algorithm 1 Pre-Processing Data

```
Require: Dataset, Count, DLC, CAN-Matrix, CAN Feature Array, Time-Series Length (TSL).
for CAN packet in Dataset: do
    if DLC < 8 then
        CAN packet ← CAN packet filling with 00;
    end if
    CAN feature packet ($1 \times 9$) ← ID and Data extract from CAN packet;
    CAN feature packet ($1 \times 19$) ← Split the CAN feature package;
    CAN feature packet ($1 \times 19$) ← feature value conversion-hexadecimal to decimal;
    Count++;
    CAN-Matrix append the CAN feature packet ($Count \times 1 \times 19$);
    if Count >= TSL then
        CAN feature Array append CAN-Matrix;
        CAN-Matrix ← CAN-Matrix remove the first element;
        Count -= 1;
    end if
end for
return CAN Feature Array
```

Based on the DLC field, the algorithm will first determine if the data field is missing. If it is indeed absent, the field "00" will be filled and the CAN packet will be updated. Next, the CAN feature packet extracts only the ID field and data field of the standard frame and splits it into 1 × 19 one-dimensional (1-D) features. Thereafter, the decimal transformed 1-D CAN feature packets are added to the CAN matrix to form 2-D CAN feature packets. In other words, we convert the dataset into 2-D spatial-temporal data X, with the t historical values of each characteristic f. The height of the matrix customization t determines how far we look into the past for future prediction. Equation (1) presents the time series of certain feature f_i:

$$T(f_i) = \{f_i(t_1), f_i(t_2), f_i(t_3), \dots, f_i(t_n)\} \tag{1}$$

where $f_i(t_j)$ denotes the ith feature of the jth time step of the multivariate time series $T(f_i)$. Thus, the time series of all 19 features can be provided in Equation (2).

$$T(f_{1,\dots,n}) = \{T(f_1), T(f_2) \cdot T(f_3), \dots, T(f_n)\} \tag{2}$$

With Equation (2), we obtain input X to the model, i.e., a multivariate tuple of temporal sequences with a custom sequence length. The custom length in this paper is 64, i.e., the input size is 64 × 19 × 1. Moreover, to balance comparability between features with a large initial range and a small initial range, initially datasets implemented z-score normalization to normalize all features. It is calculated as follows:

$$x' = \frac{x - \mu}{\sigma} \tag{3}$$

where x represents unnormalized data, x' represents normalized data, μ represents the mean value, and σ represents standard deviation of an original feature, respectively.

4.2. Two-Dimensional Temporal Convolution Network

TCN is one of the proven models to execute sequence modeling. Unlike traditional convolutional or recurrent networks, the convolution in this architecture is causal, meaning that there is no information 'leakage' from the past. In addition, the combination of residual layers and inflated convolution is used to construct very long effective history sizes. In order to apply TCN to our environment, we redesigned the original model by reducing the number of components and adjusting the parameters.

In Figure 3, we present our designed model. The overall model in Figure 3a indicates that the core component of our intrusion detection approach utilizes the residual component (ResBlock). Through the learning of normal and abnormal data, TCAN-IDS has a goal of effectively distinguishing between normal and abnormal sequences. In our approach, a single model is built for the entire system. Since the model is ultimately based on the potential features extracted to determine whether the traffic is anomalous, we perform Flatten on the deep features after three residual components. The detection phase then connects the Flattened features to the fully connected network and finally performs a two-class classification based on the Softmax function.

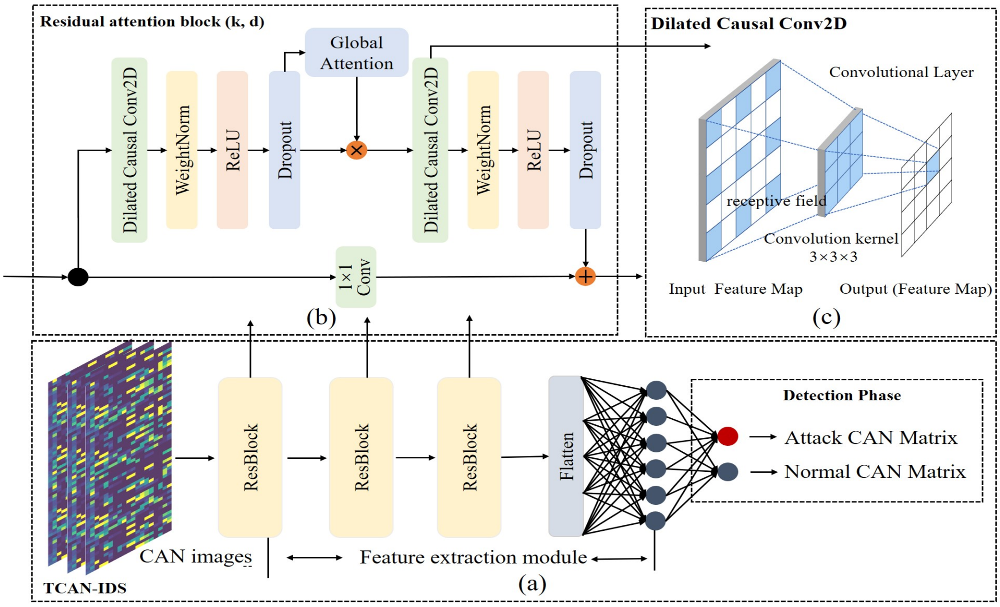

Figure 3. The structure of TCAN-IDS: (**a**) The two stages of feature extraction and detection performed by IDS on the input CAN image. (**b**) The detailed structure of the residuals in the feature extraction module. (**c**) The working model of the null convolution in the residual structure.

The detailed construction of the ResBlock is shown in Figure 3b, where a branch leading out to a series of transformations F and a 1×1 Conv branch is contained. This has repeatedly proved beneficial to very deep networks. Ultimately, the shallow features extracted by a ResBlock component are calculated as Equation (4):

$$o = \text{ Activation } (x + \mathcal{F}(x)) \tag{4}$$

where x represents the input of the ResBlock. $F(x)$ consists of three modules. Both the first and third modules consist of a dilated convolutional layer, a weight normalization layer, a rectified linear unit (ReLU), and a spatial discard layer. For normalization, we applied a weighted normalization to the convolution filters. In addition, spatial discards were added after each expanded convolution for regularization. In order to guarantee that shortcut joins and affine transforms have the same output size, our method always utilizes a 1 × 1 convolution rather than a constant mapping. The second component is a globally attentive representation of potential features, which is described in detail in the following section.

In particular, we introduce inflationary convolution in Figure 3c. For a model input with multiple time-step multivariate time series features X, dilated convolution captures important spatial-temporal detail features. Specifically, the convolution itself is able to obtain local features of X based on space. Moreover, the introduction of inflationary convolution will help the model to extract long history-dependent time-series features. That is, each col of the model input X is a 1D multivariate sequence $T(f_i)$, which is calculated by Equation (5) to convolute on element s:

$$F_d(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d\cdot i} \tag{5}$$

where d is defined as dilation factor, k is the filter size, and $s - d \cdot i$ represents the direction of the past. Apparently, the approach helps convolution to obtain a wider range of contextual relationships on temporal data. When L hidden layers are present, the receptive filed size (RFS) of a dilated convolution is computed by the following:

$$\text{RFS} = (k-1) \cdot \sum_{i=1}^{L} d_L + 1 \tag{6}$$

where d_L is the dilation factor of the N^{th} hidden layer. In order to recall more moment-to-moment information, the d-value generally grows exponentially with the depth of the network. Note that the d-value in the first hidden layer is defined as one in order to ensure that the local information is not lost in the initial input.

4.3. Attention Mechanism

Attention, as the name implies, allows it to focus on the important parts of an object. This can be effective in improving the efficiency of acquiring information [28]. The most important reason for performing this is the presence of invariant bytes in the CAN communication packet, which does not require the network to model them. Instead, CAN IDs that change frequently, and important byte changes require better attention from the model.

As shown in Figure 4, the characteristics F_1 of the first module output of the residual block are sequentially fed into the channel attention module and the spatial attention module. For the channel attention submodule, the characteristics of the inputs F_1 are permutated in three dimensions to retain information in each dimension. A two-layer MLP with shared weights then amplifies the cross-dimensional channel-space dependence. In other word, the non-linear fit of the MLP achieves an effective feature mapping of the upper level inputs, while the shared weights relieve it of the problem of too many parameters to a certain extent. Finally, reverse permutations are activated by Sigmoid to obtain channel attention coefficients $M_c(F_1)$. Channel attention features F_2 are calculated by Equation (7).

$$F_2 = M_c(F_1) \otimes F_1 \tag{7}$$

In order to focus on spatial information, F_2 is fed into the spatial attention sub-module. We use two convolutional layers to achieve spatial fusion. In addition, the channel dimensions are compressed by r. Finally, channel attention coefficients $M_s(F_2)$are generated by Sigmoid. One may note that in order to compress the data between 0 and 1, as well as to ensure constant amplitude, both the channel features and the spatial features end up with Sigmoid activation. The spatial attention features F_3 are calculated by Equation (8).

$$F_3 = M_s(F_2) \otimes F_2 \tag{8}$$

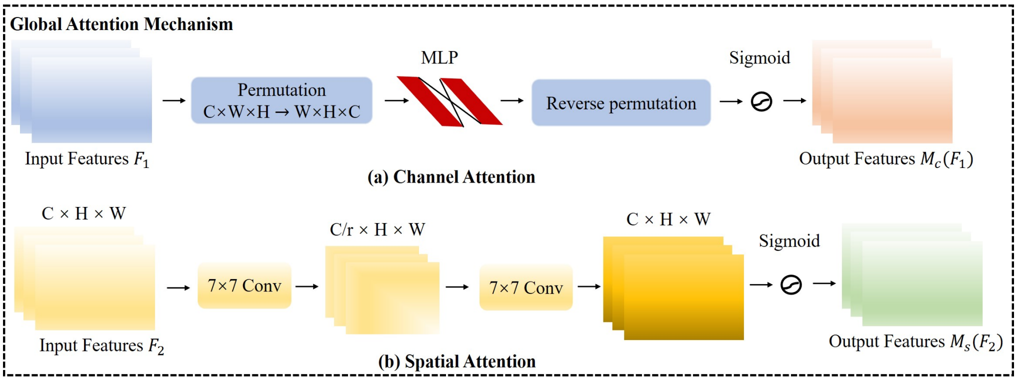

Figure 4. Global attention model structure, which includes spatial attention and channel attention. $C \times H \times W$ denotes the number of channels, feature height, and width of the feature map, respectively. The multi-layer perceptron network (MLP) learns upper layer features based on shared weights.

Thus, each residual component generates spatial-temporal detail features with global attention; the feature extraction module can take the global attention to extract valuable CAN traffic patterns.

5. Experiments Result and Discussion

Extensive experiments were conducted to discuss the performance and detection efficiency of the TCAN-IDS model.

5.1. Experimental Setup

In order to better help the model obtain more historic information, we set the TSL to 64. For per time-series 2-D images, if all 64 packets are normal traffic, that CAN image will be considered as normal image. Conversely, if there is abnormal traffic, it is considered an abnormal image. In order to avoid inflated performance due to the presence of identical attacks in the test and training data, the training dataset was divided into a 5-fold cross-validation dataset using the StratifiedKFold function in the sklearn library. In the experiment, the training set accounts for 80% of the entire dataset, while the testset accounts for 20%.

Machine Configurations. PC specification includes an Intel(R) Core (TM) i5-9300HQ CPU @ 2.40 GHz for the operating system, an NVIDIA GeForce GTX 1660 GPU and Windows 10. In order to develop algorithms, we used Python 3.6 and the Keras library. In order to validate the efficiency of the model, we employed the vehicle-class device NVIDIA Jetson AGX Xavier.

Baseline methods. We compared our TCAN-IDS method with three baseline methods as follows:

- Classification-based method. We made a direct comparison with the state-of-the-art DCNN intrusion detection algorithm proposed by Song et al. [11], where DCNN uses the arbitration bit composition 29 × 29 2-D CAN images in CAN messages to detect attacks.
- Prediction-based method. We compared this with the intrusion detection model based on the data domain proposed by Taylor et al. [32]. They are based on LSTM networks for anomaly prediction of CAN messages.
- Spatial-temporal based method. We also compared against a DeepConvGRU model, which is an ensemble of CNN and GRU model [33].

Evaluation Metrics. We evaluate the TCAN-IDS model by error rate (ER), precision, recall, and F1-Score. ER is an important metric for intrusion detection systems, and a low error rate ensures that users do not receive frequent false alarms. The accuracy rate is inversely proportional to the ER. The higher the accuracy rate, the lower the false alarm rate. It is important to capture as much abnormal behavior as possible, rather than mistaking normal behavior for abnormal behavior.

The above metrics are all calculated based on an obfuscation matrix. The four attributes are used in the confusion matrix. Based on in-vehicle intrusion detection classification results, we define these statistical metrics as follows:

1. True Positive (TP): CAN data packet is correctly classified as attacks;
2. True Negative (TN): CAN data packet is correctly classified as normal;
3. False Positive (FP): CAN data packet is incorrectly classified as attacks;
4. False Negative (FN): CAN data packet is incorrectly classified as normal.

The meaning of the five metrics is as follows:

1. Accuracy: It is the ratio of the number of correctly classified instances to the total number of instances.
$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{9}$$
2. Error rate (ER): It is the ratio of the number of incorrectly classified instances to the total number of instances.
$$\text{ER} = 1 - \text{Accuracy} = \frac{\text{FP} + \text{FN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{10}$$
3. Precision (P): It is the ratio of instances correctly classified as attacks to those actually classified as attacks.
$$P = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{11}$$
4. Recall (R): It is the ratio of attacks correctly classified as attacks to actual instances of attacks.
$$R = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{12}$$
5. F1-score: It represents a balance between precision and recall rate.
$$F = \frac{P \cdot R}{P + R} \tag{13}$$

5.2. Performance

For the training of the TCAN-IDS model, we first trained an intrusion detection model with spatial-temporal details and with global attention using normal CAN images and multiple anomalous instances of known intrusions. To demonstrate that the model is ultimately convergent and has found a globally optimal solution, we performed multiple training sessions on publicly available datasets and show training losses and accuracies for the four attack datasets in Figure 5.

From the loss of Figure 5, we observer that the spoofing attack, i.e., RPM and GEAR attacks, has the most similar loss value decreases, which corresponds to the fact that they inject the same spoofing pattern. In addition, the loss of DoS attacks is also relatively rapid, which proves that the model has learned its pattern of changing CAN communication frequencies and has a good grasp on timing characteristics. The most obvious is the fuzzy attack, which has the highest loss value and fluctuates in relation to the other three attacks. This is because the random injection of CAN ID and data increases the learning difficulty of the model.

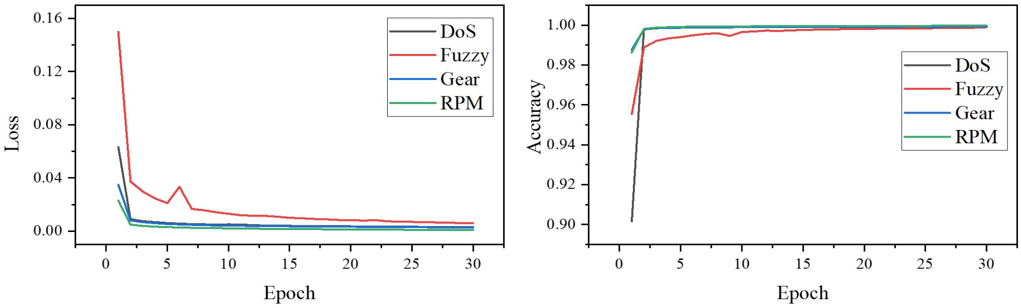

Figure 5. Loss (**left**) and Accuracy (**right**) of TCAN-IDS during training on four attack datasets.

Correspondingly, the accuracy rate reflects the same pattern. Spoofing attacks are the first to achieve the highest detection performance, with DoS attacks following closely behind. Although the fuzzing attacks are slightly less accurate than they are, they both approach 100% accuracy at 30 iterations, demonstrating the effectiveness of spatial-temporal detail and global attention.

For the field of anomaly detection, accuracy is often not the best metric to judge, Hence, TCAN-IDS also reports precision, recall, F1-score, and error rates (ER) in Table 3, where the highest performance values are highlighted in bold. In Table 3, we observed the following: TCN-IDS has higher detection precision (100%, 99.96%, 99.99%, and 99.99%), recall (99.97%, 99.89%, 99.98%, and 99.97%) and F1-score (99.98%, 99.92%, 99.98%, and 99.97%) for all four intrusion datasets compared to models based on integration (DeepConvGRU) and prediction (LSTM-P). The performance of DeepConvGRU and the LSTM-P model floated at 93% in terms of precision and recall, and it was clear that instances of false positives and false negatives could not be effectively controlled.

Table 3. TCAN-IDS performance comparison with previous work.

DoS	**ER (%)**	**P (%)**	**R (%)**	**F1 (%)**
TCAN-IDS	**0.03**	**1.0**	**99.97**	**99.98**
DCNN [11]	**0.03**	**1.0**	99.89	99.95
LSTM-P [32]	2.74	92.90	95.80	94.30
DeepConvGRU [33]	3.00	89.00	94.20	91.50
Fuzzy	**ER (%)**	**P (%)**	**R (%)**	**F1 (%)**
TCAN-IDS	**0.15**	**99.96**	**99.89**	**99.92**
DCNN [11]	0.18	99.95	99.65	99.80
LSTM-P [32]	2.90	93.30	93.30	92.80
DeepConvGRU [33]	2.60	94.20	93.00	93.90
Gear	**ER (%)**	**P (%)**	**R (%)**	**F1 (%)**
TCAN-IDS	**0.05**	**99.99**	**99.98**	**99.98**
DCNN [11]	**0.05**	**99.99**	99.89	99.94
LSTM-P [32]	4.25	91.83	90.80	91.31
DeepConvGRU [33]	3.75	92.80	91.80	92.30
RPM	**ER (%)**	**P (%)**	**R (%)**	**F1 (%)**
TCAN-IDS	**0.03**	**99.99**	**99.97**	**99.97**
DCNN [11]	**0.03**	**99.99**	99.94	99.96
LSTM-P [32]	4.25	91.80	90.80	91.30
DeepConvGRU [33]	3.75	92.81	91.68	92.24

Compared to the classification-based (DCNN) model, TCN-IDS demonstrated some advantages in fuzzy attacks, obtaining precision (99.96%), recall (99.89%), and F1 score (99.92%), respectively. Interestingly, both the TCAN-IDS model and the DCNN model use convolution to extract CAN image features, thus demonstrating the effectiveness of convolutional networks for temporal messages.

In addition, the error rate metric shows that our model has a low error rate, with ERs of 0.03%, 0.15%, 0.05%, and 0.03% on the four attack datasets, respectively. Such a result is a definite advantage over LSTM-P and DeepConvGRU. Similarly, comparable performance was achieved with the DCNN model.

5.3. Efficiency Evaluation

In this subsection, we evaluate the TCAN-IDS model detection efficiency and then demonstrate that models are available in in-vehicle environment. Table 3 presents a comparison between the TCAN-IDS model and the previous work mentioned on average detection time per message.

It is clear from Table 4 that our proposed TCAN-IDS model has a smaller time consumption than LSTM-P and DeepConvGRU. Since both CAN ID and Data domain features are considered, the speed of model inference is excessive compared to DCNN. However, the model has sufficient real-time detection capabilities relative to the transmission efficiency of the CAN bus. For example, the TCAN-IDS model can detect 64 CAN messages in 3.4 ms; that is, the model can detect 10 times the amount of CAN transmission messages in 1 s.

Table 4. The time cost of the model compared with the pervious methods.

Model	Detection Time (ms)
TCAN-IDS	3.4
DCNN [11]	0.18
LSTM-P [32]	7.6
DeepConvGRU [33]	3.17

5.4. Discussion and Limitation

Experimental results indicate that IDS based on temporal convolutional attention networks capture more spatial-temporal detailed features and valuable features, thus ensuring relatively high detection performance with a simpler structure. However, it has the following limitations:

1. The TCAN-IDS achieves excellent performance on the public injection attack dataset only but does not take into account more attacks launched by external vehicular networks.
2. In the TCAN-IDS model training phase, once it is complete, the values of k and d are fixed, making it difficult to migrate to detect more attacks.
3. Our solution is deployed on vehicle-grade devices without considering the impact of other edge computing, autonomous driving and other services that take up computing power.

6. Conclusions

Intelligent connected vehicles have been shown to be highly vulnerable to attacks, and building an effective in-vehicle intrusion detection system (IDS) has become a necessary measure. In this paper, a global attentional temporal convolutional network in-vehicle IDS is proposed to address the problems of real-time monitoring and accurate representation of anomalous features in-vehicle IDS. The model can not only focus on local features while extracting features but also extract the temporal relationship to the context. In addition, the feature mapping of important regions improves the inference speed of the model. The experimental results show that TCAN-IDS exhibits excellent performance in terms of precision, recall, and F1 score. In particular, the error rate is also kept within a reasonable range.

As the model is still a supervised approach in the training phase, it requires a large amount of data annotation, which is shown to be unrealistic. In the future, we consider using an unsupervised approach to improve the detection performance of the model. Furthermore, we will consider how to enhance the detection capabilities of TCAN-IDS for more unknown attacks.

Author Contributions: Conceptualization, M.H. and P.C.; methodology, P.C.; software, P.C.; validation, P.C.; formal analysis, K.X.; investigation, S.L.; resource, M.H.; data curation, P.C.; writing—original draft preparation, P.C.; writing—review and editing, M.H. and K.X.; visualization, K.X.; supervision, M.H.; project administration, M.H.; funding acquisition, M.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the following funds: Key Research and Development Plan of Jiangsu province in 2017 (Industry Foresight and Generic Key Technology) (BE2017035); Project of Jiangsu University Senior Talents Fund (1281170019).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Pawelec, K.; Bridges, R.A.; Combs, F.L. Towards a CAN IDS based on a neural network data field predictor. In Proceedings of the ACM Workshop on Automotive Cybersecurity, Richardson, TX, USA, 27 March 2019; pp. 31–34.
2. Qin, H.; Yan, M.; Ji, H. Application of Controller Area Network (CAN) bus anomaly detection based on time series prediction. *Veh. Commun.* **2021**, *27*, 100291. [CrossRef]
3. Checkoway, S.; McCoy, D.; Kantor, B.; Anderson, D.; Shacham, H.; Savage, S.; Koscher, K.; Czeskis, A.; Roesner, F.; Kohno, T.; et al. *Comprehensive Experimental Analyses of Automotive Attack Surfaces*; USENIX Security Symposium: San Francisco, CA, USA, 2011; Volume 4, p. 2021.
4. Tariq, S.; Lee, S.; Kim, H.K.; Woo, S.S. CAN-ADF: The controller area network attack detection framework. *Comput. Secur.* **2020**, *94*, 101857. [CrossRef]
5. Jhong, S.Y.; Chen, Y.Y.; Hsia, C.H.; Lin, S.C.; Hsu, K.H.; Lai, C.F. Nighttime object detection system with lightweight deep network for internet of vehicles. *J. Real-Time Image Process.* **2021**, *18*, 1–15. [CrossRef]
6. Aliwa, E.; Rana, O.; Perera, C.; Burnap, P. Cyberattacks and countermeasures for in-vehicle networks. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–37. [CrossRef]
7. Miller, C.; Valasek, C. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA* **2015**, *2015*, S91.
8. Fowler, D.S.; Bryans, J.; Shaikh, S.A.; Wooderson, P. Fuzz testing for automotive cyber-security. In Proceedings of the 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), Luxembourg, 25–28 June 2018; pp. 239–246.
9. Rouf, I.; Miller, R.D.; Mustafa, H.A.; Taylor, T.; Oh, S.; Xu, W.; Gruteser, M.; Trappe, W.; Seskar, I. *Security and Privacy Vulnerabilities of In-Car Wireless Networks: A Tire Pressure Monitoring System Case Study*; USENIX Security Symposium: San Francisco, CA, USA, 2010; Volume 10.
10. Jaw, E.; Wang, X. Feature Selection and Ensemble-Based Intrusion Detection System: An Efficient and Comprehensive Approach. *Symmetry* **2021**, *13*, 1764. [CrossRef]
11. Song, H.M.; Woo, J.; Kim, H.K. In-vehicle network intrusion detection using deep convolutional neural network. *Veh. Commun.* **2020**, *21*, 100198. [CrossRef]
12. Bangui, H.; Buhnova, B. Recent Advances in Machine-Learning Driven Intrusion Detection in Transportation: Survey. *Procedia Comput. Sci.* **2021**, *184*, 877–886. [CrossRef]
13. Roshani, M.; Sattari, M.A.; Ali, P.J.M.; Roshani, G.H.; Nazemi, B.; Corniani, E.; Nazemi, E. Application of GMDH neural network technique to improve measuring precision of a simplified photon attenuation based two-phase flowmeter. *Flow Meas. Instrum.* **2020**, *75*, 101804. [CrossRef]
14. Sanaat, A.; Zaidi, H. Depth of interaction estimation in a preclinical PET scanner equipped with monolithic crystals coupled to SiPMs using a deep neural network. *Appl. Sci.* **2020**, *10*, 4753. [CrossRef]
15. Sattari, M.A.; Roshani, G.H.; Hanus, R.; Nazemi, E. Applicability of time-domain feature extraction methods and artificial intelligence in two-phase flow meters based on gamma-ray absorption technique. *Measurement* **2021**, *168*, 108474. [CrossRef]
16. Azizi, A.; Pleimling, M. A cautionary tale for machine learning generated configurations in presence of a conserved quantity. *Sci. Rep.* **2021**, *11*, 1–10. [CrossRef] [PubMed]
17. Alotaibi, Y. A New Database Intrusion Detection Approach Based on Hybrid Meta-Heuristics. *CMC Comput. Mater. Contin.* **2021**, *66*, 1879–1895.
18. Shrestha, R.; Omidkar, A.; Roudi, S.A.; Abbas, R.; Kim, S. Machine-learning-enabled intrusion detection system for cellular connected UAV networks. *Electronics* **2021**, *10*, 1549. [CrossRef]
19. Subramani, N.; Mohan, P.; Alotaibi, Y.; Alghamdi, S.; Khalaf, O.I. An Efficient Metaheuristic-Based Clustering with Routing Protocol for Underwater Wireless Sensor Networks. *Sensors* **2022**, *22*, 415. [CrossRef]

20. Yang, L.; Moubayed, A.; Shami, A. MTH-IDS: A Multi-Tiered Hybrid Intrusion Detection System for Internet of Vehicles. *IEEE Internet Things J.* **2021**, *9*, 616–632. [CrossRef]
21. Kang, M.J.; Kang, J.W. Intrusion detection system using deep neural network for in-vehicle network security. *PLoS ONE* **2016**, *11*, e0155781. [CrossRef]
22. Zhong, M.; Zhou, Y.; Chen, G. Sequential model based intrusion detection system for IoT servers using deep learning methods. *Sensors* **2021**, *21*, 1113. [CrossRef]
23. Sun, H.; Chen, M.; Weng, J.; Liu, Z.; Geng, G. Anomaly Detection for In-Vehicle Network Using CNN-LSTM With Attention Mechanism. *IEEE Trans. Veh. Technol.* **2021**, *70*, 10880–10893. [CrossRef]
24. Tariq, S.; Lee, S.; Woo, S.S. CANTransfer: transfer learning based intrusion detection on a controller area network using convolutional LSTM network. In Proceedings of the 35th Annual ACM Symposium on Applied Computing, Gwangju, Korea, 30 March–3 April 2020; pp. 1048–1055.
25. Zhang, C.; Song, D.; Chen, Y.; Feng, X.; Lumezanu, C.; Cheng, W.; Ni, J.; Zong, B.; Chen, H.; Chawla, N.V. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 1409–1416.
26. Yue, C.; Wang, L.; Wang, D.; Duo, R.; Nie, X. An Ensemble Intrusion Detection Method for Train Ethernet Consist Network Based on CNN and RNN. *IEEE Access* **2021**, *9*, 59527–59539. [CrossRef]
27. Bai, S.; Kolter, J.Z.; Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv* **2018**, arXiv:1803.01271.
28. Liu, Y.; Shao, Z.; Hoffmann, N. Global Attention Mechanism: Retain Information to Enhance Channel-Spatial Interactions. *arXiv* **2021**, arXiv:2112.05561.
29. Yue, C.; Wang, L.; Wang, D.; Duo, R.; Yan, H. Detecting Temporal Attacks: An Intrusion Detection System for Train Communication Ethernet Based on Dynamic Temporal Convolutional Network. *Secur. Commun. Netw.* **2021**, *2021*, 3913515. [CrossRef]
30. Seo, E.; Song, H.M.; Kim, H.K. Gids: Gan based intrusion detection system for in-vehicle network. In Proceedings of the 2018 16th Annual Conference on Privacy, Security and Trust (PST), Belfast, Ireland, 28–30 August 2018; pp. 1–6.
31. Han, M.; Cheng, P.; Ma, S. PPM-InVIDS: Privacy protection model for in-vehicle intrusion detection system based complex-valued neural network. *Veh. Commun.* **2021**, *31*, 100374. [CrossRef]
32. Taylor, A.; Leblanc, S.; Japkowicz, N. Anomaly detection in automobile control network data with long short-term memory networks. In Proceedings of the 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Montreal, QC, Canada, 17–19 October 2016; pp. 130–139.
33. Zhang, J.; Wu, Z.; Li, F.; Xie, C.; Ren, T.; Chen, J.; Liu, L. A deep learning framework for driving behavior identification on in-vehicle CAN-BUS sensor data. *Sensors* **2019**, *19*, 1356. [CrossRef]